

System informatyczny dla wieloetapowego rozpoznawania obiektów

Computer system for multi-stage object recognition

Swietłana Lebediewa¹

Treść. Przedmiotem pracy jest przedstawienie informatycznego systemu wspomagającego proces wieloetapowego podejmowania decyzji w procesie wieloetapowego rozpoznawania obiektów.

Przedstawiono ideę rozpoznawania wieloetapowego jako metodę polegającą na dekompozycji problemu decyzyjnego. Przedstawiono charakterystykę drzewa decyzyjnego i ciągu uczącego. Zaprezentowano bazę wiedzy i operacyjną bazę danych do rozpoznawania wieloetapowego. Omówiono współpracę algorytmów rozpoznawania z bazą danych.

Słowa kluczowe: baza danych, baza wiedzy, rozpoznawanie wieloetapowe.

Abstract. The subject of this paper is to present the specificity of computer system for systems of multistage decision making in the process multistage pattern recognition. The paper presents the idea of multi-stage recognition as a method involving the decomposition of the decision problem. The characteristics of the decision tree and the learning sequence are presented. Knowledge base and operational database to a multi-step recognition are presented.

Cooperation recognition algorithms to the database are discussed.

Keywords: database, multi-stage recognition, knowledge base.

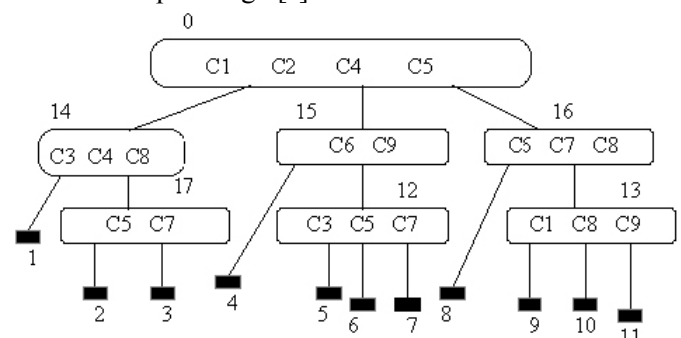
1. Wstęp

W procesie podejmowania złożonych decyzji znaczne miejsce znajdują metody wykorzystujące podejścia polegające na dekompozycji problemu decyzyjnego. Dzięki takim dekompozycjom zazwyczaj uzyskujemy możliwość przetwarzania większych problemów decyzyjnych. Jednym ze stosowanych podejść jest podejmowanie wieloetapowych decyzji z wykorzystaniem drzewa decyzyjnego jako „szkieletu” prowadzenia zdekomponowanego procesu decyzyjnego. Przykładem takiego postępowania jest rozpoznawanie wieloetapowe, w którym zastępujemy jednorazowe rozpoznawanie sekwencją tzw. rozpoznawań lokalnych, których przeprowadzenie jest zgodne z zadaną konstrukcją drzewa decyzyjnego, tj. wzdłuż jednej z możliwych ścieżek korzeń-węzeł terminalny. Takie podejście pozwala uzyskać miarodajne rozpoznanie badanego obiektu, zazwyczaj z efektywniejszym niż w przypadku rozpoznawania jednorazowego, wykorzystaniem danych pomiarowych oraz pozostałych zasobów odpowiedniego systemu komputerowego. Metodyka rozpoznawania wieloetapowego została wprowadzona w pracy [1,3].

2. Rozpoznawanie wielostopniowe

Rozpoznawanie wielostopniowe (RW) obiektów [1, 2, 3] polega na tym, że w poszczególnych etapach rozpoznawania

podjmuje się decyzję, do którego podzbioru wszystkich możliwych klas należy klasa rozpoznawanego obiektu oraz o tym, jakie cechy ze zbioru wszystkich możliwych cech mają być mierzone w następnym etapie rozpoznawania. Tak więc rozpoznawanie wielostopniowe polega na *dekompozycji* problemu decyzyjnego, czyli zastępowaniu jednorazowego rozpoznawania sekwencją tzw. rozpoznawań lokalnych, przeprowadzanych w poszczególnych węzłach zgodnie z zadaną konstrukcją **drzewa decyzyjnego** (DD), ryc.1.1. Żeby zaklasyfikować rozpoznawany obiekt do jednej z klas, należy przejść ścieżką w DD, startując od korzenia. W każdym napotykanym węźle wewnętrznym (**decyzyjnym**) należy podjąć decyzję o dalszej drodze w grafie, aż osiągnięty zostanie węzeł terminalny. Klasa z nim związana reprezentuje końcowy wynik rozpoznawania wielostopniowego [4].



Ryc. 1. 1. Drzewo decyzyjne

Algorytmy rozpoznawania wielostopniowego (ARW) z uczeniem korzystają z ciągu uczącego (CU), ryc. 1.2. CU jest ciągiem par postaci (x_k, k) , gdzie x_k jest wektorem wartości cech obiektu, k zaś – numerem klasy [2,3], symbol * w CU oznacza, że cecha jest nieistotna dla klasy i przy obliczaniu „odległości” nie jest brana pod uwagę. W dalszych rozważaniach będziemy rozpatrywać współpracę z CU *Algorytmu najbliższego sąsiada* (ANS). Algorytm ten polega na tym, że rozpoznawany obiekt klasyfikowany jest do tej samej klasy, do której należy element CU, którego „odległość” od rozpoznawanego obiektu jest najmniejsza („odległość” jest pewną normą w przestrzeni cech).

| C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | KLASA |
|----|------|----|------|------|----|-----|-----|------|-------|
| 16 | 3,44 | 3 | 6,55 | 40,0 | 48 | 116 | * | 14,9 | 7 |
| 15 | 3,45 | 2 | 8,45 | 40,0 | * | * | 120 | * | 1 |
| 14 | 2,4 | 1 | 8,33 | 38,1 | * | 110 | * | * | 2 |
| 18 | 3,0 | * | 7,5 | 35,0 | 40 | * | * | 15,0 | 4 |
| 15 | 2,7 | * | 6,0 | 20,0 | 35 | 110 | 112 | * | 8 |

Ryc. 1.2. Fragment ciągu uczącego.

3. System informatyczny dla rozpoznawania wielostopniowego

System informatyczny dla rozpoznawania wielostopniowego zawiera elementy bazy wiedzy (BW), algorytmy rozpoznawania i operacyjną bazę danych (BD). Operacyjną BD przedstawiono w paragrafach 3 i 4, algorytmy rozpoznawania – w paragrafach 3 i 5.

Elementy bazy wiedzy to:

- Zbiór poprawnie sklasyfikowanych obiektów (zbiór faktów doświadczalnie zebranych i zweryfikowanych) reprezentowany przez elementy CU - wiedza pochodząca od eksperta;
- Drzewo decyzyjne (klasyfikator oparty na DD) pozwalające zdekomponować proces rozpoznawania na etapy - wiedza pochodząca od eksperta;
- Algorytmu rozpoznawania - wiedza pochodząca od eksperta.

CU ma postać tablicy prostokątnej, w ostatniej kolumnie, w której znajdują się numery klas obiektów, w pozostałych – wartości cech poprawnie sklasyfikowanych obiektów, ryc. 1.2.

Drzewo decyzyjne może być przedstawione jako spójny, nieorientowany, acykliczny graf skończonego stopnia (drzewa), z każdym węzłem którego jest związana pewna informacja, wykorzystywana przez algorytmy rozpoznawania. Informacja ta może być przedstawiona w postaci tablic, ryc.2.1 [5,6]

Tablica a) zawiera informacje o liczbie cech wykorzystywanych w węźle i liczbie bezpośrednich następników węzła. Tablica b) przyporządkowuje każdemu z węzłów nieterminalnych numery cech wykorzystywanych w tym węźle. Tablica c) przyporządkowuje węzłom nieterminalnym numery klas osiągalnych z tych węzłów. Tablica d) zawiera informację o numerach następników każdego węzła nieterminalnego wraz z informacją, czy następnik jest

węzłem terminalnym czy nie.

a)

| NR WĘZŁA | LICZBA CECH | LICZBA NASTĘPNIKÓW |
|----------|-------------|--------------------|
| 0 | 4 | 2 |
| 14 | 3 | 2 |
| 11 | 2 | 2 |
| 15 | 2 | 2 |
| 12 | 3 | 3 |

b)

| CECHA | WĘZŁ |
|-------|------|
| C1 | 0 |
| C2 | 0 |
| C3 | 14 |
| C4 | 0 |
| C5 | 0 |
| C6 | 12 |

c)

| WĘZŁ | KLASA |
|------|-------|
| 17 | 2 |
| 17 | 3 |
| 14 | 1 |
| 14 | 2 |
| 14 | 3 |
| 12 | 5 |
| 12 | 6 |
| 15 | 4 |
| 15 | 5 |

d)

| WĘZŁ | NASTĘPNIK | WĘZŁ TERMINALNY |
|------|-----------|-----------------|
| 0 | 14 | 0 |
| 0 | 15 | 0 |
| 14 | 1 | 1 |
| 14 | 17 | 0 |
| 17 | 2 | 1 |
| 17 | 3 | 1 |
| 15 | 4 | 1 |
| 15 | 12 | 0 |

Ryc. 2.1. Tablice opisujące strukturę DD.

Opisu drzewa decyzyjnego dostarcza ekspert. Ekspert określa również algorytmy rozpoznawania pozwalające na podstawie wartości zmierzonych cech zaliczyć obiekt do tej czy innej klasy.

4. Segment danych i algorytm rozpoznawania w węźle

Do podjęcia decyzji w węźle algorytm rozpoznawania potrzebuje następującej informacji: informacji o fragmencie CU i cechach obiektu wykorzystywanych w tym węźle, informacji o numerach klas osiągalnych z tego węzła, informacji o tym, czy wynik rozpoznawania w węźle (numer klasy) jest następnikiem danego węzła, a jeżeli nie, informacji, z którego węzła-następnika danego węzła klasa ta jest osiągalna. Informacja potrzebna do podjęcia decyzji w jednym węźle drzewa decyzyjnego znajduje się w jednym zbiorze danych. Zbiór taki nazywa się **segmentem** a identyfikator segmentu indeksuje się numerem i węzła decyzyjnego, z którym segment ten jest związany. Na każdym etapie rozpoznawania wieloetapowego algorytmy rozpoznawania współpracują z jednym segmentem danych, pobierając informację zawartą w segmencie i po przetworzeniu tej informacji, wpisując wyniki przetwarzania (podjęte decyzje) do segmentu. Tak więc segment danych stanowi autonomiczną jednostkę bazy danych do rozpoznawania wieloetapowego, a dla rozpoznawania jednoetapowego cała baza danych redukuje się do jednego segmentu. Niech symbol * oznacza numer węzła. Segment zawiera następującą informację [5]: Informację o cechach wykorzystywanych w węźle* (CECHY*), informację o następnikach węzła* (NASTĘPNIKI*), informację o klasach osiągalnych z węzła* (KLASY_OSIĄGALNE*), informację o fragmencie CU używanym w węźle* (FRAGSEQ*), informację o wartościach cech mierzonych w węźle* (TAB*), a także informację o decyzjach podjętych w tym węźle (DECYZJE*). Decyzją może być numer klasy lub numer węzła nieterminalnego, do którego trze-

ba przekazać sterowanie. W celu ilustracji przedstawimy bazę danych zredukowaną do jednego segmentu – relacyjnej BD zredukowanej do segmentu 14 na ryc.3.1.

NASTĘPNIKI_14

| NR_WĘZŁA | NR_WĘZŁA NASTĘPNIKA | T |
|----------|---------------------|-----|
| 14 | 1 | TAK |
| 14 | 17 | NIE |

CECHY_14

| NR_CECHY | NR_WĘZŁA |
|----------|----------|
| C3 | 14 |
| C4 | 0 |
| C8 | 14 |

KLASY_OSIĄGALNE_14

| NR_KLASY | NR_DECYZJI | T |
|----------|------------|-----|
| 1 | 1 | TAK |
| 2 | 17 | NIE |
| 3 | 17 | NIE |

FRGSEQ_14

| C3 | C4 | C8 | NR_KLASY |
|----|------|-----|----------|
| 3 | 6,58 | 128 | 1 |
| 4 | 6,31 | 145 | 3 |
| 1 | 6,12 | 137 | 3 |
| 2 | 6,75 | 186 | 2 |
| 5 | 6,43 | 129 | 1 |
| 3 | 6,13 | 135 | 2 |
| 2 | 6,02 | 112 | 1 |

DECYZJE 14 (fragment)

| O-ID | NR_WĘZŁA | T |
|------|----------|-----|
| 001 | 1 | TAK |
| 005 | 17 | NIE |
| 007 | 17 | NE |

TAB14 (fragment)

| O-ID | C3 | C4 | C8 |
|------|----|------|-----|
| 001 | 2 | 6,24 | 125 |
| 005 | 1 | 6,86 | 131 |
| 007 | 1 | 6,35 | 145 |

Ryc.3.1 Model bazy danych zredukowanej do segmentu SEG14

ALGORYTM ROZPOZNAWANIA W WĘZŁE* (symbol * oznacza numer węzła)

- KROK 1.** Wczytaj z BD identyfikator i wektor cech rozpoznawanego obiektu.
- KROK 2.** Wczytaj fragment CU FRAGSEQ* wykorzystywany w danym węźle.
- KROK 3.** (obliczanie wartości funkcji rozpoznawania wektora cech rozpoznawanego obiektu **g**): dla wszystkich elementów relacji FRAGSEQ*: pobierz element CU, oblicz normę euklidesową z rozpoznawanym obiektem, zapamiętaj normę euklidesową i numer klasy obiektu, której ona odpowiada.
- KROK 4.** Zapisz identyfikator obiektu i rezultat rozpoznawania w WĘZŁE* do relacji DECYZJE* w BD. **STOP.**

5. Model konceptualny i modele zewnętrzne bazy danych dla RW

Najprostszym rozwiązaniem jest przyjęcie modelu konceptualnego będącego sumą segmentów danych. Gdyby

wszystkie zapytania do BD były związane z AR - wędrówką rozpoznawanego obiektu po DD - potraktowanie modelu konceptualnego jako sumy segmentów byłoby nie tylko naturalne ale i optymalne. W każdym segmencie łatwo zrealizować odpowiedzi na zapytania:

Podaj wektor cech wykorzystywany w węźle *;

Jakie klasy są osiągalne w węźle *;

Jakie decyzje podjęto w węźle * o obiekcie NR-Obiektu;

Jakie następniki ma węzeł *;

Do jakiego węzła należy przekazać sterowanie, jeżeli rezultatem decyzji jest klasa *i*;

Jaki węzeł jest poprzednikiem węzła *;

Ile następników ma węzeł *;

Podaj wszystkie cechy obiektu NR-Obiektu i rezultat rozpoznawania w węźle *.

Natomiast, jeżeli do BD REC będą kierowane zapytania nie związane bezpośrednio z AR, np. zapytania dotyczące struktury DD, przyjęcie modelu konceptualnego jako sumy segmentów byłoby kłopotliwe. Rzeczywiście, realizacja zapytań typu:

Podaj wektor cech DD;

Jaki jest poprzednik klasy *i*;

Podaj węzły na ścieżce (węzeł-1, węzeł-2);

Jakie obiekty należą do klasy *i*?

Wymagałoby całkowitego przeglądu BD. Dlatego przy projektowaniu modelu konceptualnego brano pod uwagę zarówno realizację zapytań związanych z rozpoznawaniem jak i realizację zapytań dotyczących struktury DD. Tak więc w modelu konceptualnym bazy danych REC występują relacje opisujące strukturę drzewa decyzyjnego (relacje WĘZŁY, KLASY, NASTĘPNIKI i KLASY OSIĄGALNE), relacja CECHY zawierająca informację o cechach wykorzystywanych w poszczególnych węzłach i informację o tym, w którym węźle cecha była mierzona po raz pierwszy, relacja SEQUENCE opisująca ciąg uczący, relacje typu TAB zawierające wartości cech rozpoznawanych obiektów, relacje typu DECYZJE zawierające rezultaty decyzji podejmowanych w poszczególnych węzłach, a także relacja SEGMENTY zawierająca informację o relacjach występujących w poszczególnych segmentach. Relacja WĘZŁY przyporządkowuje numerom węzłów nieterminalnych nazwy segmentów danych, a także zawiera taką informację o węzłach nieterminalnych, jak liczba cech wykorzystywanych w tym węźle, liczba bezpośrednich następników węzła i liczba jego poprzedników. Na ryc 4.1 przedstawiono fragment schematu bazy danych. Klucze relacji zaznaczono symbolem #.

RELATION WĘZŁY(NR_WĘZŁA#, SEG_ID, LICZBA_CECH, BEZPOŚREDNIE-NASTĘPNIKI, POPRZEDNIKI, BEZ_POPRZEDNIK)

RELATION SEGMENTY (SEG_D#, NR_WĘZŁ, CU, NAZWA_TAB, CECHY, NASTĘPNIKI, DECYZJE, KLASY_OSIĄGALNE)

RELATION KLASY (KLASA#, NR_WĘZŁA)

RELATION NASTĘPNIKI (NR_WĘZŁA#, NR_WĘ-

ZŁA_POPRZEDNIK)
RELATION KLASY OSIĄGALNE (KLASA#, NR_WĘZŁA#)
RELATION CECHY (CECHA#, NR_WĘZŁA#, PIERWSZE WYSTĄPIENIE)
RELATION SEQUENCE (C1#,C2#,C3#,C4#,C5#,C6#,C7#,C8#,C9#, KLASA)
RELATION TAB 0 (OB_ID#,C1,C2, , C4, C5)
RELATION DECYZJE 0 (OB_ID#, NR_WĘZŁA, T)

Ryc. 4.1. Schemat bazy danych dla rozpoznawania wieloetapowego (fragment)

Relacja KLASY opisuje zależność pomiędzy numerem klasy a numerem węzła będącego bezpośrednim poprzednikiem tej klasy, dla każdej klasy wskazany jest jej poprzednik – Ryc.4.2a,. Relacja NASTĘPNIKI zawiera informacje o bezpośrednich następnikach węzłów nieterminalnych (i informację o bezpośrednich poprzednikach węzłów nieterminalnych będących wartościami atrybutu NR_WĘZŁA_NASTĘPNIK) – Ryc. 4.2b. Relacje WĘZŁY, KLASY i NASTĘPNIKI łącznie opisują strukturę drzewa decyzyjnego. Relacje KLASY, KLASY OSIĄGALNE i NASTĘPNIKI, pozwalają określić dowolną ścieżkę w drzewie decyzyjnym. W relacji CECHY zawarta jest informacja, w których węzłach wykorzystywane są poszczególne cechy i w którym węzle cecha jest mierzona po raz pierwszy.

Relacja typu DECYZJE zawiera rezultaty decyzji podejmowanych przez algorytmy rozpoznawania dla kolejno rozpoznawanych obiektów. Relacja DECYZJE* (* jest numerem węzła nieterminalnego) ma trzy atrybuty: OB_ID, NR_WĘZŁA i T - identyfikator obiektu, numer węzła będącego rezultatem decyzji podjętej w węzle * oraz informacja, czy rezultatem decyzji jest węzeł terminalny (klasa) czy nie. Informacja o rozpoznawanych obiektach znajduje się w relacjach typu TAB*, fragmenty relacji TAB14 i relacji DECYZJE14 oraz fragment relacji CECHY zredukowanej do cech wykorzystywanych w węzle 14 przedstawiono na Ryc. 3.1. Informacja o CU znajduje się w relacji SEQUENCE.

a) Relacja KLASY

| KLASA | NR_WĘZŁA |
|-------|----------|
| 1 | 14 |
| 2 | 17 |
| 3 | 17 |
| 4 | 15 |
| 5 | 12 |
| 6 | 12 |
| 7 | 12 |
| 8 | 16 |
| 9 | 13 |
| 10 | 13 |
| 11 | 13 |

b) Relacja NASTĘPNIKI

| NR_WĘZŁA_NASTĘPNIKA | NR_WĘZŁA |
|---------------------|----------|
| 17 | 14 |
| 12 | 15 |
| 13 | 16 |
| 14 | 0 |
| 15 | 0 |
| 16 | 0 |

Ryc.4.2a, b. Model konceptualny. Relacje KLASY i NASTĘPNIKI

c) Relacja KLASY OSIĄGALNE

| NR_KLASA | NR_WĘZŁA |
|----------|----------|
| 2 | 17 |
| 3 | 17 |
| 1 | 14 |
| 2 | 14 |
| 3 | 14 |
| 4 | 15 |
| 5 | 15 |
| 6 | 15 |
| 7 | 15 |
| 5 | 12 |
| 6 | 12 |
| 7 | 12 |
| 8 | 16 |
| 9 | 16 |
| 10 | 16 |
| 11 | 16 |
| 9 | 13 |
| 10 | 13 |
| 11 | 13 |
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0 |
| 10 | 0 |
| 11 | 0 |

Ryc 4.2c. Model konceptualny. Relacja KLASY OSIĄGALNE

Oprócz modelu konceptualnego w bazie danych istnieją modele zewnętrzne – fragmenty BD, z którymi pracuje użytkownik - jeden lub kilka segmentów danych.

6. Współpraca algorytmów rozpoznawania wieloetapowego z bazą danych

Przed rozpoczęciem rozpoznawania trzeba dokonać otwarcia sesji, BD i wszystkich relacji modelu konceptualnego. Oznaczmy przez C_N – bieżący węzeł, przez C_SEG_ID – bieżący identyfikator segmentu, R_N niech oznacza numer węzła–korzenia DD a D_N – numer węzła będącego rezultatem rozpoznawania. Liczbę rozpoznawanych obiektów oznaczamy zmienną W. Przedstawimy ARW współpracujący z BD:

KROK 0.

OPEN S; OPEN DB; OPEN R; READ R_N.
 [Otwarcie sesji, BD i wszystkich relacji modelu konceptualnego. Odczytanie z BD numeru węzła–korzenia DD]. C_N:=RN; COUNTER:=0.

KROK 1.

CREATE SEGMENT; [Utworzenie segmentu (model zewnętrzny BD) dla aktualnego węzła]

KROK 2. **OPEN R_SEG** [otwarcie wszystkich relacji w segmencie].

KROK 3. **PROJECT INFSEG* ON LICZBA_CECH GIVING C_LICZBA_CECH**
SELECT TAB* WHERE O_ID=C_O_ID GIVING C_TAB;
PROJECT C_TAB ON O_ID GIVING C_ID.

KROK 4. **CALL ALGORYTM ROZPOZNAWANIE-W_WĘZLE*.**

KROK 5. **SELECT DECYZJE* WHERE O_ID=C_O_ID GIVING C_DEC; PROJECT C_DEC ON WĘZEL GIVING D_N.**

KROK 6. **IF D_N="węzeł terminalny" THEN GOTO KROK 8;**

KROK 7. **C_N=D_N; GOTO KROK1.**

KROK 8. **COUNTER:=COUNTER+1; C_N=R_N; IF (COUNTER<w) THEN GOTO KROK 1 ELSE CLOSE S; STOP.**

W kroku 0 następuje otwarcie sesji, bazy danych i wszystkich relacji modelu konceptualnego oraz nadanie wartości początkowych zmiennym. W kroku 1 zostaje utworzony segment danych dla aktualnego węzła: Zostają utworzone relacje **INFSEG***, **WĘZEL***, **CECHY***, **NASTĘPNIKI***, **KLASY_OSIĄGALNE***, **FRAGSEQ***. W kroku 2 zostają otwarte wszystkie utworzone relacje segmentu **SEG*** a także relacje **TAB*** i **DECYZJE***. W kroku 3 zostaje pobrana informacja o liczbie cech wykorzystywanych w węźle *. W kroku 4 wywołuje się algorytm rozpoznawania w węźle. W kroku 5 następuje pobranie z relacji **DECYZJE*** wiersza, zawierającego identyfikator rozpoznawanego obiektu i numer węzła (podjętej decyzji), następnie zapamiętanie decyzji w zmiennej **D_N**. W następnych krokach sprawdza się, czy rezultat rozpoznawania jest węzłem terminalnym (co oznacza koniec rozpoznawania) i pobiera się następny element do rozpoznawania. Jeżeli rezultat rozpoznawania jest węzłem nieterminalnym, to ten węzeł (rezultat rozpoznawania) czyni się węzłem aktualnym i do niego zostaje przekazane sterowanie.

Operacje występujące w algorytmach rozpoznawania można podzielić na dwie grupy: instrukcje języka relacyjnej bazy danych i instrukcje realizujące funkcję rozpoznawania (np. obliczające "odległość" (normę euklidesową) rozpoznawanego obiektu od elementów CU i wybierającą najmniejszą z tych "odległości"). Złożoność obliczeniową funkcji rozpoznawania dla ANS można oszacować w sposób następujący. Przyjmijmy, że długość wektora cech w węźle * równa się l . Do obliczenia metryki używa się jednej operacji odejmowania, jednej operacji mnożenia, $(l-1)$ operacji dodawania i jednej operacji wyciągnięcia pierwiastka kwadratowego, razem $3*l$ operacji arytmetycznych. Następnie wykonuje się $(m-1)$ operacji porównania (m – długość **FRAGSEQ***). Ostatecznie do obliczenia funkcji rozpoznawania w węźle potrzeba $3*l*(m-1)$ operacji. Najbardziej czasochłonną operacją z pierwszej grupy jest operacja tworzenia modelu zewnętrznego (**CREATE**

SEGMENT). Oznaczmy przez n długość (liczbę wierszy) najdłuższej relacji modelu konceptualnego. Niech * oznacza numer węzła. Relacje **CECHY***, **NASTĘPNIKI***, **WĘZEL*** i **INFSEGMENT*** powstają z odpowiednich relacji modelu konceptualnego za pomocą operacji **SELECT**. Złożoność obliczeniowa każdej z tych operacji jest $O(n)$. Relacja **FRAGSEQ*** powstaje z relacji **SEQUENCE** przez superpozycję operacji **SELECT** i **PROJECT**. Złożoność obliczeniową operacji **PROJECT** jest $O(m^2)$, gdzie m liczba wierszy relacji będącej argumentem operacji **PROJECT**. Niech s będzie długością najdłuższej ścieżki w drzewie. Wtedy algorytm rozpoznawania dla jednego obiektu wymaga s operacji **CREATE SEGMENT**, s -krotnego wywołania **ALGORYTMU ROZPOZNAWANIA W WĘZLE***, oraz wykonania kilkunastu relacyjnych operacji. Dla $s=m$ pesymistyczna złożoność obliczeniowa jest więc $O(m^2+m)$, czyli – $O(m^2)$. **ALGORYTM ROZPOZNAWANIA W WĘZLE*** pracuje na modelu zewnętrznym, co daje znaczną oszczędność czasu. Zamiast porównywać zmierzone cechy rozpoznawanego obiektu z każdym elementem CU, porównuje się je z elementami fragmentu CU **FRAGSEQ***, który jest krótszy od CU. Oznaczmy przez K liczbę węzłów terminalnych DD (liczbę klas), przez LE – liczbę elementów CU dla jednej klasy, a przez NK_i – liczbę klas osiągalnych z węzła nieterminalnego o numerze i . Współpraca algorytmu rozpoznawania z CU **SEQUENCE** występującym w modelu konceptualnym wymaga $K*LE$ porównań wektora cech rozpoznawanego obiektu z elementami CU, a współpraca z CU **FRAGSEQ*** występującym w modelu zewnętrznym wymaga NK_i*LE porównań. Dla jednego tylko węzła oszczędność wynosi $(K - NK_i)*LE$ porównań. Przykładowo, przy założeniu, że liczba elementów CU dla każdej klasy równa się 10, praca z całym CU dla drzewa z ryc.2.1 wymaga $11*10 = 110$ porównań w każdym węźle, a praca z **FRAGSEQ*** wymaga 110 porównań w węźle 0, 30 porównań w węźle 14 i 20 porównań w węźle 17. Oszczędność dla każdego obiektu z klasy 1 na ścieżce (0,14) wynosi $110*2 - (110+30) = 80$, a na ścieżce (0,14,17) – $110*3 - (110+30+20) = 170$ porównań.

7. Uwagi końcowe

System ekspertowy jest systemem komputerowym używającym modeli wiedzy i procedur wnioskowania w celu rozwiązywania problemów o dużej skali trudności [7]. Podstawowa idea działania systemów ekspertowych polega na przeniesieniu wiedzy eksperta z określonej dziedziny do bazy wiedzy i posiadaniu posiadających informacji zaprojektowaniu maszyny wnioskującej [8]. Pełna realizacja takiego systemu pozwala umiejscowić go w klasie systemów ekspertowych przeznaczonych do wspomaganiania wypracowania zautomatyzowanej decyzji diagnostycznej [9]. Jednym z podstawowych problemów przy konstrukcji takiego systemu ekspertowego jest stworzenie i eksploatacja jego bazy wiedzy z faktami (CU, DD) i zaprojektowaniu maszyny wnioskującej realizującej algorytm rozpoznawania.

Literatura (References)

- [1] Z. Bubnicki, *Knowledge-Based Approach as a Generalization of Pattern Recognition Problems and Methods*. *Systems Science*, 1993, Vol. 19, No 2, pp. 5-21.
- [2] M. Fłasiński, *Wstęp do sztucznej inteligencji*, PWN, Warszawa 2011.
- [3] M. Kurzyński, *Algorytmy rozpoznawania wieloetapowego oraz ich zastosowania medyczne i techniczne*, Wyd. PWr. ,Wrocław 1987.
- [4] J. Józefczyk, *Rozpoznawanie i zastosowania biomedyczne*, [w:] *Problemy automatyki i informatyki*, Wrocław: Wyd. Ossolineum 1998, 45-58.
- [5] S. Lebediewa, B. Węglorz, *Problemowo zorientowane bazy danych w procesie wielostopniowego podejmowania decyzji*, [w:] *Współczesne problemy informatyki*, Legnica 2011, 127-142.
- [6] S. Lebediewa, *Metodologia projektowania problemowo zorientowanych baz danych do systemów wielostopniowego podejmowania decyzji*, Wyd. PWr. 1998.
- [7] J. Kisielnicki, H. Sroka, *Systemy informatyczne biznesu*, Warszawa 2005.
- [8] L. Rutkowski, *Metody i techniki sztucznej inteligencji*, Warszawa, Wyd. PWN 2009.
- [9] J. Józefczyk, *Systemy z reprezentacją wiedzy. Systemy ekspertowe*, [w:] *Problemy automatyki i informatyki*, Wrocław, Wyd. Ossolineum 1998, 71-82.